Claims:

We claim:

1. A computer operating system comprising a kernel, the kernel configured to encrypt and decrypt data transferred between a computer memory and a secondary device.

2. The computer operating system of claim 1, wherein the kernel comprises an encryption engine configured to encrypt clear data to generate cipher data, the encryption engine further configured to decrypt the cipher data to generate the clear data.

3. The computer operating system of claim 2, further comprising a memory portion coupled to the encryption engine and configured to store the cipher data.

4. The computer operating system of claim 2, wherein the encryption engine is configured to encrypt clear data and decrypt cipher data according to a symmetric key encryption algorithm.

5. The computer operating system of claim 4, wherein the symmetric key encryption algorithm is based on a block cipher.

6. The computer operating system of claim 5, wherein the symmetric key encryption algorithm comprises the Rijndael algorithm.

7. The computer operating system of claim 6, wherein the symmetric key encryption algorithm uses a block size of 128 bits, 192 bits, 256 bits, 512 bits, 1024 bits, or 2048 bits.

8. The computer operating system of claim 6, wherein the symmetric key encryption algorithm uses a key length of 128 bits, 192 bits, 256 bits, 512 bits, 1024 bits, or 2048 bits.

9. The computer operating system of claim 5, wherein the symmetric key encryption algorithm comprises a DES algorithm.

10. The computer operating system of claim 5, wherein the symmetric key encryption algorithm comprises a Triple-DES algorithm.

11. The computer operating system of claim 5, wherein the symmetric key encryption algorithm comprises an algorithm selected from the group consisting of IDEA, Blowfish, Twofish, and CAST-128.

12. The computer operating system of claim 1, wherein the kernel comprises a UNIX operating system.

13. The computer operating system of claim 12, wherein the UNIX operating system is a System V-Revision.

14. The computer operating system of claim 3, wherein the memory portion comprises a first logical protected memory configured to store encrypted file data and a second logical protected memory configured to store encrypted key data.

15. The computer operating system of claim 14, further comprising an encryption key management system, the encryption key management system configured to control access to the encrypted file data and the encrypted key data.

16. The computer operating system of claim 15, wherein the encryption key management system comprises a key engine, the key engine configured to receive a pass key and the file name to generate an encrypted file name key, the key engine further configured to use the encrypted file name key and file contents to generate an encrypted file contents key, the key engine further configured to encrypt the file contents with the encrypted file contents key to generate encrypted file contents.

17. The computer operating system of claim 16, wherein the encryption key management system is configured to store encrypted file names, wherein the file names are associated with the encrypted file contents.

18. The computer operating system of claim 17, wherein the encryption key management system is further configured to grant access to a file if a corresponding access permission of the file is a predetermined value.

19. The computer operating system of claim 18, wherein the secondary device is accessed using a file abstraction.

20. The computer operating system of claim 19, wherein the secondary device is a backing store.

21. The computer operating system of claim 19, wherein the secondary device is a swap device.

22. The computer operating system of claim 19, wherein the secondary device is a socket connection.

23. The computer operating system of claim 22, wherein the socket connection comprises a computer network.

24. The computer operating system of claim 23, wherein the computer network comprises the Internet.

25. The computer operating system of claim 17, wherein the encryption key management system is further configured to encrypt the pathname to the encrypted data, the encryption key management system further configured to decrypt the pathname to the encrypted data when retrieving encrypted file contents.

26. A computer system comprising:

a. a first device having an operating system kernel, the operating system kernel configured to encrypt clear data using an encryption key to generate cipher data, the first device further configured to decrypt the cipher data using the encryption key to generate the clear data; and

b. a second device coupled to the first device and configured to exchange cipher data with the first device.

27. The computer system of claim 26, wherein the operating system kernel is configured to encrypt the clear data and decrypt the cipher data using a symmetric algorithm.

28. The computer system of claim 27, wherein the symmetric algorithm comprises a block cipher.

29. The computer system of claim 28, wherein the block cipher comprises a Rijndael algorithm.

30. The computer system of claim 29, wherein the encryption key comprises at least 1024 bits.

31. The computer system of claim 26, wherein the second device comprises a backing store.

32. The computer system of claim 26, wherein the second device comprises a swap device.

33. The computer system of claim 26, wherein the second device comprises a communications channel.

34. The computer system of claim 33, wherein the communications channel comprises a network.

35. The computer system of claim 34, wherein the network comprises the Internet.

36. A method of encrypting data, the method comprising:

a. receiving clear data; and

b. executing kernel code in an operating system, the kernel code using a symmetric key to encrypt the clear data to generate cipher data, the kernel code further using the symmetric key to decrypt the cipher data to generate the clear data.

37. The method of claim 36, wherein the symmetric key encrypts the clear data to generate cipher data according to a block cipher.

38. The method of claim 37, wherein the block cipher comprises a Rijndael algorithm.

39. The method of claim 37, wherein the block cipher comprises an algorithm selected from the group consisting of DES, triple-DES, Blowfish, and IDEA.

40. The method of claim 36, wherein executing kernel code comprises:

a. entering a pass key and a file name into a first encryption process to produce an encrypted file name and an encrypted file name key; and

b. processing the file contents with the encrypting file name key to generate an encrypted file contents key and an encrypted file contents.

41. The method of claim 40, further comprising:

a. storing the encrypted file name key and the encrypted file contents key in a first protected area of a computer storage; and

b. storing the encrypted file name and the encrypted file contents in a second protected area of the computer storage.

42. The method of claim 36, wherein executing kernel code to encrypt clear data and decrypt cipher data is performed when data is transferred between a computer memory and a secondary device.

43. The method of claim 42, wherein the secondary device comprises a backing store.

44. The method of claim 42, wherein the secondary device comprises a swap device.

45. The method of claim 42, wherein the secondary device comprises a communications channel.

46. The method of claim 45, wherein the communications channel comprises a network.

47. The method of claim 46, wherein the network comprises the Internet.

48. A computer system comprising:

a. a processor;

b. a physical memory;

c. a secondary device coupled to the physical memory; and

d. an operating system comprising a kernel, the kernel configured to encrypt and decrypt data transferred between the physical memory and the secondary device.

49. The computer system of claim 48, wherein the kernel is configured to encrypt and decrypt data using a symmetric key encryption algorithm.

50. The computer system of claim 49, wherein the symmetric key encryption algorithm is based on a block cipher.

51. The computer system of claim 50, wherein the symmetric key encryption algorithm comprises the Rijndael algorithm.

52. The computer system of claim 51, wherein the kernel comprises a UNIX operating system.

53. A method of accessing a file, the method comprising:

a. authenticating a user;

b. checking the user's permission to access the file; and

c. encrypting the file using an encryption key.

54. The method of claim 53, wherein encrypting the file comprises:

a. dividing the file into a plurality of file segments, each file segment having an associated file segment number;

b. dividing each file segment into a plurality of corresponding file blocks;

c. dividing the encryption key into a plurality of corresponding encryption key segments;

d. permutating the corresponding encryption key segments using the associated file segment number and a first permutation function to produce a corresponding intermediate key;

e. encrypting the corresponding file blocks using an encryption algorithm and the corresponding intermediate key to generate a corresponding first encrypted data; and

f. permutating the corresponding first encrypted data using a second permutation function and the associated file number to generate corresponding final encrypted data.

55. The method of claim 54, wherein the encryption algorithm comprises the Rijndael algorithm.

56. The method of claim 53, wherein the first permutation function differs from the second permutation function.

57. The method of claim 53, wherein each file segment is at least 1024-bits long.

58. The method of claim 53, wherein the encryption key is at least 2048-bits long.